DOCUMENT RESUME

ED 416 822 IR 018 717

| | |
|---|---|
| TITLE | Detailed Evaluation of a Novel Approach to Curricular Software. |
| INSTITUTION | Tufts Univ., Medford, MA. |
| SPONS AGENCY | Fund for the Improvement of Postsecondary Education (ED), Washington, DC. |
| PUB DATE | 1994-00-00 |
| NOTE | 20p. |
| CONTRACT | P116B11580 |
| PUB TYPE | Reports - Descriptive (141) -- Reports - Evaluative (142) |
| EDRS PRICE | MF01/PC01 Plus Postage. |
| DESCRIPTORS | *Academic Achievement; *Active Learning; *Computer Assisted Instruction; Computer Software Evaluation; *Courseware; Curriculum Evaluation; Experimental Curriculum; Higher Education; *Instructional Effectiveness; *Statistics |
| IDENTIFIERS | Fund for Improvement of Postsecondary Education |

ABSTRACT

This report describes a detailed, multi-site evaluation of ConStatS, a curricular software package for helping students conceptualize introductory statistics. ConStatS employs a novel approach for forcing students into an active, experimental style of learning. This evaluation allowed an assessment of the degree to which curricular software helps a much wider range of students to adopt an experimental style of learning and whether their doing so brings them closer in performance to superior students. In the process, the research was designed to be a paradigm for evaluating curricular software in general with a large sample of courses and studies, and with state-of-the-art assessment procedures. The project was conducted during the period September 1, 1991-December 31, 1994. Overall, 20 introductory statistics courses at five colleges and universities participated in the evaluation. Sixteen of the classes used the software and four did not. Students in all classes were given a test of statistical concepts contained both in the software and taught in the classes that did not use the software. With 103 concepts tested, students using the software did better on 94 of the 103 questions. Though the software benefited all students who used it, students with remedial problems in basic mathematics showed the smallest overall gain. Results of the project offer university statistics teachers an estimate of gains they can expect from a curricular reform offered by ConStatS. The methods developed to evaluate ConStatS are appropriate for other technology-based learning programs. The evaluation was also useful for learning how to diagnose which students will benefit from software by the way they use it. (SWC)

Final Report for: Detailed Evaluation of a Novel Approach to Curricular Software

Grantee Organization:

Tufts University
Psychology Department and the Software Studio
Medford, MA 02155

Grant Number:

P116B11580

Project Dates:

Starting Date: September 1, 1991
Ending Date: December 31, 1994
Number of Months: 40

Project Director:

Richard Chechile
Psychology Department
Tufts University
Medford, MA 02155
Telephone: (617) 627-3765

FIPSE Program Office: David Johnson

| Grant Award: | | |
|---|---|---|
| | Year 1 | $ 67,066 |
| | Year 2 | $ 70,188 |
| | Year 3 | $ 57,726 |
| | Total | $ 194,980 |

## SUMMARY

This report describes a detailed, multi-site, evaluation of ConStatS, a curricular software package for helping students conceptualize introductory statistics. ConStatS employs a novel approach for forcing students into an active, experimental style of learning. This evaluation allowed us to assess the degree to which curricular software helps a much wider range of students to adopt an experimental style of learning and whether their doing so brings them closer in performance to superior students. In the process, the research is designed to be a paradigm for evaluating curricular software in general with a large sample of courses and studies and with state-of-the-art assessment procedures.

## PROJECT TEAM

Richard Chechile, George Smith, Steve Cohen
Curricular Software Studio
Arena User Area
Tufts University
Medford, MA 02155

## PROJECT REPORTS AND PUBLICATIONS

Cohen, S., Chechile, R., Smith, G., Tsai, F., Burns, G., (1994). A Method for Evaluation Instructional Software. Behavior Research Methods,Instruments, and Computers, 26 236-241.

Cohen, S., Smith, G., Chechile, R., Cook, B., (1994). Designing Software for Conceptualizing Statistics, Proceedings of the First Conference of the International Association for Statistics Education

## IN PRESS

Cohen, S., Chechile, R., Smith, A Detailed, Multisite Evaluation of Curricular Software, In Assessment Strategies that Work, forthcoming from Jossy-Bass, San Francisco

Cohen, S., Tsai, F., Chechile, R. A (1995). Method for Assessing Student Interaction with Instructional Software, In Behavior Research Methods,Instruments, and Computers, 27,

## IN REVISION

Cohen, S., Smith, G., Chechile, R., & Burns, G., Impediments to Learning Probability and Statistics from an Assessment of Curricular Software

Cohen, S., & Chechile, R., Probability Distributions, Assessment & Instructional Software: Lessons Learned from an Evaluation of Curricular Software

Cohen, S., et al., ConStatS: Software for Conceptualizing Statistics

## EXECUTIVE SUMMARY

1. Project Name: Detailed Evaluation of a Novel Approach to Curricular Software

2. Address: Curricular Software Studio
Arena User Area
Tufts University
Medford, MA 02155

3. Contacts: Professor Richard A. Chechile, (617) 627 - 3765
Steve Cohen, (617) 627 - 3082, SCohen@emerald.tufts.edu
Professor George Smith, (617) 628 - 5000, x2344

### A. PROJECT OVERVIEW

Under a prior grant from FIPSE, the Software Studio had developed ConStatS, a set of programs for helping students develop a deep conceptual understanding of introductory statistics. The software had a unique interface that forced students to pose questions and execute focused experiments about specific concepts. The principal goal of this project was to learn if ConStatS was successful, and if software that emphasized an active, experimental style of learning would help a large fraction of students in mathematics and science courses develop a deep understanding of the material.

The first year of this three year project was spent developing a method for assessing the effectiveness of ConStatS. The goal was to develop a method that would provide detailed results about ConStatS and could also be applied to a wide range of technology-based learning programs. The second and third years of the project were used to evaluate the software at a variety of sites. Overall, twenty introductory statistics courses at five different colleges and universities participated in the evaluation. Sixteen of the classes used the software and four did not. Students in all classes were given a test of statistical concepts contained both in the software and taught in the classes that did not use the software. In all, 103 concepts were tested. Students using the software did better on 94 of the 103 questions. Though the software did benefit all students who used it, those students with remedial problems in basic mathematics showed the smallest overall gain.

The project served several audiences. The results offer statistics teachers at the university level a detailed estimate of what gains they can expect from the kind of curricular reform offered by ConStatS. They also have implications for the kind of learning style necessary to succeed in other courses with substantial theoretical components (e.g. chemistry, physics, calculus). The methods developed to evaluate ConStatS are appropriate for other technology-based learning programs, especially large scale programs designed to improve conceptual understanding of mathematics and science. Finally, the evaluation was useful not only for evaluating the software, but also for learning how to diagnose which students will benefit from software by the way they use it.

### B. PURPOSE

We see the project having three main purposes. First, goal was to obtain solid data on our claim that an "active, experimental" approach is indispensable to learning subjects dominated by abstract theories. Talk of "active learning" has become a cliche. Our claim concerns a very specific form of active learning, namely devising and carrying through systematic thought experiments. Outstanding students invariably seem to do this. The vast majority of students do not. Second, the goal was to obtain detailed information about how curricular software ought to be designed. For example, the basic claim is that properly designed curricular software will enable a much larger fraction of students to adopt the approach to learning followed by the most successful students. The underlined phrase is crucial. We found during the development of ConStatS that even so much as a vestige of didactic instruction or educational television in the software was

enough to nudge students back into their usual mode of learning, e.g. asking what lesson they were supposed to be coming away with. Finally, the third goal concerns the evaluation process itself. We have two concerns about methods and practices for evaluating curricular software. First, too many evaluations focus on such things as speed of learning and student attitude rather than on depth of comprehension and retention. Second, almost no evaluations attempt to identify exactly what does and does not work for students of different types. The long range impact of curricular software ultimately depends on our learning how to design it in detail to achieve specific ends. In other words, like any new technology, educational innovations have to go through an "engineering development" over time. ConStatS puts us in a position to show what can be gained from a proper evaluation.

C. Background and Origins

ConStatS development was funded by FIPSE from 1987 through 1990. Introductory statistics is probably the least successful course in higher education. Few students leave the course with any ability to reason statistically. That problem alone motivated ConStatS. From the outset, it was not clear how best to design software to address this problem. A first generation of ConStatS which also required students to learn by active experimentation was much too open-ended. All but a small fraction of students, left to their own devices, either froze or became lost when they had to make all the decisions. The second generation of ConStatS employed a unique combination of devices to solve this problem. First, each program in the package is divided into a large number of "screens", no one of which confronts the student with more than a small number of closely related decisions. The choices the student makes on each screen lead to different pathways through the program, pathways that often loop into one another. Each screen has a one or two sentence "scaffolding" introducing the choices that have to be made, and the student can always back up along a pathway to review or reconsider earlier choices. Finally, and most important, a WHY and HELP button are always available on every screen, allowing the students who are uncertain access to information that will help them over the hurdle.

The program had been used at Tufts for two years before the evaluation and reached a level of polish both technically and pedagogically. It was clear that students could use the program easily, but it was not clear just how extensive the benefits were -- it was not clear if the software offered a sufficient enough improvement for all students that introductory statistics courses should be reorganized to take full advantage of it.

D. Project Description

"Is ConStatS effective?" Achieving a detailed and reliable answer to the question is a very complicated matter. Running a tightly control study in one or two courses would not tell us much about learning outcomes and how they would transfer to the vast array of statistics courses offered in higher education. Factors unique to the course (e.g. text, instructor, select course topics) would limit the applicability of the results. (This shortcoming is common in evaluations of instructional technology.) To accomplish the evaluation and have the results be meaningful to a wide range of universities, colleges, and community colleges, the evaluation needed to be executed as a field study with many different sites, each using ConStatS as it best fit their existing curricula. Consequently, we developed and installed a facility in the program for recording how each student and class used the software. This facility permitted us to account for differences in performance based on differences in use. In addition, ConStatS is an extensive program that address most of the concepts taught in an introductory statistics course. A single measure of success (e.g. a test score) would not indicate precisely which portions of the program were successful and which were not. In order to obtain detailed information that would inform both the evaluation and the redesign of the software, it was essential to identify which concepts were taught by each part of the program and develop a question to test each concept. The detailed testing along with records of program use were sufficient for a field study approach.

E. Evaluation

Our results strongly suggest that curricular software makes a clear difference, but students with a poor or limited mathematical background will not achieve the gains that more prepared students might. Using a short 10 question pretest we were able to identify those students whose limited mathematics skills did not permit them get the full benefit from the software. It should be noted that ConStatS does not address remedial problems.

As for evaluating the project itself, we sought feedback about our own work through formal conference presentations and publications. Presenting work to a critical audience is often the best way to assess your own work. Conference presentations were very well received, often ending with listeners wanting to use some or all of the methods presented. One of the papers on the evaluation has been published in a refereed journal, while another has been included in a book on "Assessment Strategies That Work". We look forward to critical comments on our work that is currently out for review.

F. Results and Conclusions

We learned an extraordinary amount about ConStatS, active experimental learning, evaluation and teaching statistics. This kind of evaluation should accompany every large scale educational innovation.

# FINAL REPORT TO FIPSE

## A. PROJECT OVERVIEW

Under a prior grant from FIPSE, the Curricular Software Studio at Tufts had developed ConStatS, a set of programs for helping students develop a deep conceptual understanding of introductory statistics. The software was developed to help students comfortably assume an active, experimental style of learning that we believed was essential for developing a deep conceptual understanding of statistics and other abstract disciplines. Formative evaluations of early versions of the software showed us that all but the best students had problems posing questions and executing meaningful experiments -- most students froze when they were forced to make a choice or interpret a result. Consequently, the software was redesigned. Programs were broken into sets of screens, each one of which forced students to make a choice or execute a focused experiment. A new set of formative evaluations took place and the program was field tested and used for two years at Tufts. Several different statistics classes used the software. These new formative evaluations showed that the software was essentially free of bugs and that students could use the software and experiment easily. Once it was clear the software was technically robust and easy to use, two new questions emerged: 1) exactly what were the students doing with the software and 2) would the software help students develop a better conceptual understanding of statistics.

The first year of this three year project was spent developing a method for evaluating outcomes and student use. The goal was to develop methods that would provide detailed results about ConStatS and that could also be applied to a wide range of technology-based learning programs. In order to assess what students did and did not learn from ConStatS we had to first define precisely what ConStatS was trying to teach. This was a difficult task. ConStatS, like almost all instructional technology programs that emphasize an experimental style of learning, have an opened nature. There are many options, each of which leads to a different educational experience. To address this problem, we designed a method for systematically defining each part

of a program, recording the set of possible exercises, and then describing the (comprehension) points a student might be expected to learn by engaging in the experiment. This method yield over 1000 comprehension points, each of which was linked to a particular part of the program. Since we could not test so many separate points, we looked for identical or similar points and defined clusters of comprehension points. Each cluster was defined by a common underlying concept. Ultimately we defined 103 concepts, each of which could be tested by a well targeted question. This method for defining the domain of concepts taught by ConStatS should be transferable to other programs as well.

In addition to defining a method for identifying concepts, it was import to learn what parts of the programs students used and how they used them. If students did not use a particular part of the program, it would not be reasonable to expect them to learn the associated concept(s). It they did use a particular part, then it became important to learn what they did (i.e., what kind of interactions with the program proved effective and which did not). We designed a taxonomy of interactions, each of which had a unique educational objective (i.e., execute an experiment, ask for help, study a result). A system for recording the interactions was installed in the software. The system should also be transferable to other instructional technology programs.

The second and third years of the project were used to evaluate the software at a variety of sites. Overall, twenty introductory statistics courses at five different colleges and universities participated in the evaluation. Sixteen of the classes used the software and four did not. Students in all classes were given a test of statistical concepts contained both in the software and taught in the classes that did not use the software. In all, 103 concepts were tested. Students using the software did better on 94 of the 103 questions. Though the software did benefit all students who used it, those students with remedial problems in basic mathematics showed the smallest overall gain. The summer between the second and third was used year to make changes to the software base on second year evaluation results. Fifteen significant changes were made, but these changes did not yield improved results. Thus while the assessment demonstrated which parts of the program worked well and which did not, the results were not useful for redesigning those parts of

the program that were not effective. In one case, improved integration into the curriculum (i.e., improved assignments, better examples) did result in improved performance. Finally, the records of student interactions showed that some student constructed better, more meaningful experiments than others. Those that performed better experiments tended to better on related questions.

The project served several audiences. The results offer statistics teachers at the university level a detailed estimate of what gains they can expect from the kind of curricular reform offered by ConStatS. They also have implications for the kind of learning style necessary to succeed in other courses with substantial theoretical components (e.g. chemistry, physics, calculus). The methods developed to evaluate ConStatS are appropriate for other technology-based learning programs, especially large scale programs designed to improve conceptual understanding of mathematics and science. Finally, the evaluation was useful not only for evaluating the software, but also for learning how to diagnose which students will benefit from software by the way they use it.

## B. PURPOSE

Our intent in proposing the project was to use ConStatS as an instrument to get at other, more basic issues. Hence, should the findings not generalize, the project will be a failure in our own eyes. We foresee it generalizing in three distinct ways.

First, we wanted to obtain data about on our claim that an "active, experimental" approach is indispensable to learning subjects dominated by abstract theories. Talk of "active learning" has become a cliche. Our claim concerns a very specific form of active learning, namely devising and carrying through systematic thought experiments. Outstanding students invariably seem to do this. The vast majority of students do not. One might think that the tendency to engage in thought experiments is a result of having learned the material. We think, to the contrary, that it is the principal means of gaining conceptual control of the material. The project gave us an opportunity to assess this thesis. ConStatS is ultimately nothing but a prosthesis for conducting thought experiments. The traces we employed in the evaluation project gave us a record of the specific experiments each student tried in each part of the software. Thus we know not only the topics on

which they did and did not engage in a number of experiments, but also such things as whether they tried several variations of certain experiments. By systematically correlating these data with the learning demonstrated by the individual students on each topic, we can pinpoint the role thought experiments played in their mastery of each part of the subject matter, and thereby test the general hypothesis about learning.

Second, a goal of the project was to obtain detailed information about how curricular software ought to be designed. For example, our basic claim is that properly designed curricular software will enable a much larger fraction of students to adopt the approach to learning followed by the most successful students. The underlined phrase is crucial. We found during the development of ConStatS that even so much as a vestige of didactic instruction or educational television in the software was enough to nudge students back into their usual mode of learning, e.g. asking what lesson they were supposed to be coming away with. We have tried to remove all such vestiges from ConStatS, but we surely have not succeeded. The project goal, therefore, was to reveal which parts of the software work well with students of various sorts, and which parts do not.

In addition to such general conclusions about curricular software design, another project goal was also to learn about the effectiveness a particular novel software-interface features adopted in ConStatS -- using WHYs and HELPs. Once the publisher of our software package on human anatomy saw the WHYs and HELPs in ConStatS, he asked us to incorporate them in that software as well. No single feature of ConStatS is attracting more immediate attention among reviewers than the WHY and HELP device is; those reviewing the package at IBM, for example, have been especially taken with it. In short, we think that the WHY and HELP device can and will be used quite widely in curricular software. But we were still proceeding mostly on the basis of hunches in deciding what sort of specific content is needed in the WHYs and HELPs. Consequently, a project goal was to obtain solid data on which WHYs and HELPs work and which do not, and hence be in a position to draw general conclusions about how this device is best implemented.

One further point ought to be made about what the project intends to reveal about curricular software generally. From the day Tufts Curricular Software Studio was founded, our dream has

been one of helping a much wider range of students achieve the levels of proficiency in so-called "hard" courses that only the best students now achieve. We see little point in making superior students still better. If curricular software is to be at all revolutionary, it must make a difference to the students who now struggle with demanding material. We are especially hopeful that curricular software can make a difference for educationally disadvantaged students. The obvious problem here is to determine whether any initial failure to make such a difference for these students is a consequence of readily correctable gaps in their background knowledge or a consequence of deeper factors that curricular software cannot easily reach.

Finally, another goal for the project was to be an example of good software evalution. For fear of offending reviewers, we were reluctant to be as outspoken in the proposal about how poorly curricular software and other new educational technologies were being evaluated. We had two complaints. First, too many evaluations focus on such things as speed of learning and student attitude rather than on depth of comprehension and retention. Second, almost no evaluations attempted to identify exactly what does and does not work for students of different types. The long range impact of curricular software ultimately depends on our learning how to design it in detail to achieve specific ends. In other words, like any new technology, educational innovations have to go through an "engineering development" over time. Detailed data on what is and is not working are indispensable to such engineering. In the absence of thorough evaluations of the sort done in this project, the promise of curricular software is liable to slip through our hands in much the way the promise of educational television has, where the second and third generations show little improvement on the first. We wanted the project to become the benchmark for evaluating curricular software.

C. Background and Origins

ConStatS development was funded by FIPSE from 1987 through 1990. Introductory statistics is probably the least successful course in higher education. Few students leave the course with any ability to reason statistically. That problem alone motivated ConStatS. From the outset,

it was not clear how best to design software to address this problem. A first generation of ConStatS used a point and click, graphical user interface. We hoped the "user friendly interface" would permit students to easily construct insightful experiments. While many student could point and click, few could construct effective experiments. The software was much too open-ended. All but a small fraction of students, left to their own devices, either froze or became lost when they had to make all the decisions. The second generation of ConStatS employed a unique combination of devices to solve this problem. First, each program in the package is divided into a large number of "screens", no one of which confronts the student with more than a small number of closely related decisions. The choices the student makes on each screen lead to different pathways through the program, pathways that often loop into one another. Each screen has a one or two sentence "scaffolding" introducing the choices that have to be made, and the student can always back up along a pathway to review or reconsider earlier choices. Finally, and most important, a WHY and HELP button are always available on every screen, allowing the student, who is uncertain or hesitant, access to information that will help them over the hurdle. It wasn't until this second generation of ConStatS was complete, and we were convinced that the program genuinely supported active, experimental learning, that a detailed evaluation made sense.

The development team and the formative evaluations also influenced the large scale evaluation design. Tufts has at least eight separate introductory statistics courses in different academic departments. Each course, though it might be called introductory statistics, was very different. Faculty from each of these departments consulted on the design of ConStatS. The software had to meet the needs of each teacher. Consequently, as each member of the design team integrated early versions of the software into their courses, it became clear that no two teachers used it the same way. Each emphasized different parts of the program. Some used assignments while others gave students free reign. The early trial use demonstrated that a good evaluation would require a record of how the program was used. It also became evident that a single teacher would exhaust the program and several different sites that emphasize different parts of the program would be needed. This insight forced us to consider a multivariate assessment measure that would

capture the effectiveness of each part of the program at different sites. Finally, as each teacher seemed to require flexible use, we realized that a set of carefully controlled randomized experiments would not work. This forced us to adopt a quasi-experimental, program evaluation model that employed detailed testing and extensive implementation monitoring. This flexible evaluation model made it easier to attract faculty at outside sites to participate in the evaluation. No school turned down the opportunity to participate because the evaluation guidelines were too restrictive. Had we not gone through the design and formative evaluation ourselves, we may have still taken a program evaluation approach to help achieve generalizable results. However, the emphasis on monitoring student use and extensive multivariate assessment came from the development years.

Finally, Tufts had a decision to make when ConStatS was complete. The same decision is facing almost all colleges and universities. About 300 students or more a semester take introductory statistics at Tufts. Making a commitment to a package meant redesigning many courses and providing enough computer facilities to support the student load comfortably. (The introductory psychology statistics course at Tufts requires its own dedicated lab with 18 networked machines). Teaching assistants had to be trained on ConStatS as well. All this required an enormous investment. It was clear that students could use the program easily, but it was not clear just how extensive the benefits were -- it was not clear if the software offered enough of an improvement that all introductory statistics courses should be reorganized to take full advantage of it. Faculty needed reasons to make changes and central administrators needed evidence that the investment in computer facilities would be a good value.

D. Project Description

"Is ConStatS effective?" Achieving a detailed and reliable answer to the question is a very complicated matter. We did not try an execute a classic experiment to answer our question. It is unlikely we could have successfully executed a **semester long** experiment by splitting a class in two and having half the class use ConStatS while the the other half recived an alternative

curriculum for improving conceptual understanding. Student study groups could not be policed and students would have likely shared ideas. It would have been nearly impossible to develop an alternative curriculum to ConStatS that emphasized conceptual understanding. There are a number of possible alternatives -- either emphasizing active experimentation without technology or technology without active experimentation.

Even if we did succeed in running a tightly controlled study in one or two courses, it would not tell us much about how learning outcomes would transfer to the vast array of statistics courses offered in higher education. The evaluation would not have been a success if the results did not generalize beyond one or two courses at Tufts. If only one or two evaluation sites were used, factors unique to the course (e.g. text, instructor, selected course topics) or the institution (kind of students, facilities, etc.) would severely limit the applicability of the results. (This shortcoming is common in evaluations of instructional technology.) To accomplish the evaluation and have the results be meaningful to a wide range of universities, colleges, and community colleges, the evaluation needed to include many sites and disciplines.

Hence, we executed the evaluation as a field study with many different sites, each using ConStatS as it best fit their existing curricula. Faculty inside of Tufts were relatively easy to recruit since they were already using parts of the software and were curious about their own investment. Sites outside of Tufts were a bit harder to secure. The hope was to find several sites different from Tufts, and one that was similar (to guard against the possibility that the program only works effectively at the home institution). Ultimately we succeeded in recruiting a state school (University of New Hampshire), a school with a unique population (Gallaudet), a school where students take one course at a time (Colorado College), and a school similar to Tufts (Bowdoin). A member of the team executing the evaluation met with each faculty member participating in the evaluation to introduce the software and explain the goals of the evaluation. We hoped to have an urban state university (to compare with UNH), and did recruit one school, UMass Boston. They were unable to execute the evaluation. Finally, despite repeated efforts, no community college ever joined the study. A single community college does not offer as many statistics courses as a typical

four-year school, thus reducing the possibilities within a school. Several community colleges that showed interest lacked the technical resources to participate.

In addition to recruiting sites, the first year was used to develop (1) a pretest of requisite skills necessary for students to use ConStatS, (2) a detailed test of conceptual understanding that assessed every part of the program, and (3) a facility in the program for recording how each student and class used the software. The test of conceptual understanding required a detailed content analysis of over two hundred ConStatS screens. The analysis yielded over 1000 comprehension points taught by the software. Rather than try to test the 1000 points (an impossible task), each point was compared with the others in an effort to join similar or identical point into common clusters. This clustering exercise yielded 103 concepts. A question testing conceptual understanding was drafted and reviewed by two outside statistical experts and teachers of quantitative methods.

The facility for recording interaction first required an educational interaction analysis of the software. The kind of analysis yielded a description of the kinds of interactions offered by the software and maps them to the educational purpose that motivated them. Once there was a complete set of interactions defined and mapped to an educational taxonomy, the goal was to install a facility in the software that would capture each interaction, interpret it, and create a data base of all student interactions with the software. Engineering the software tracing facility required a sophisticated programming effort by three students. Ultimately, it permitted us to account for differences in performance based on differences in student usage.

Before the evaluation began there was a one day workshop for all faculty participating. The goals and requirements were reviewed, and the options for curriculum integration were discussed. Individual faculty joining the evaluation during the second and third years were given one-on-one assistance.

The second year was spent executing the evaluation, analyzing data, and trying to use the results to improve the program. Executing and monitoring the evaluation meant bi-weekly communication with sites. Site visits were done to install the program, solve technical problems,

administer the comprehesion tests, and collect data. The evaluation generated an enormous amount of useful data. The trace facility generated about 40 million bytes of detailed data on how students used the program. Analysis of the second year data suggested several areas in the software where improvements were necessary. It was only possible to designed and implement 15 of these changes during the summer between the second and third years. The additions and changes to the software were programmed by three students during the summer, and a new set of evaluations were initiated at the start of the third year.

E. Evaluation

Since our project was an evaluation, it is important to consider two issues. One, what did our evaluation tell us. Secondly, how did we evaluate our own evaluation.

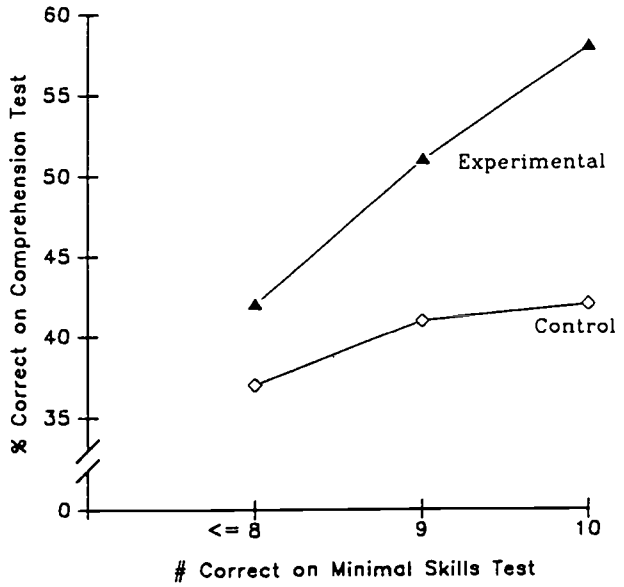
Our results to date strongly suggest that curricular software makes a big difference. Overall, students using the program did better on 94 out of 103 questions. The increase does not seem to be owed to increased learning time, as at least two classess substituted a computer lab for a class. The improvement does not seem to be tied to a particular method for integrating the program into a curriculum, as several teachers with different approaches achieved relatively comparable results.

However, students with a poor or limited mathematical background did not achieve the gains that more prepared students might. ConStatS does not address remedial problems or teach students the basic mathematical skills neccessary to use software. All students participating in the evaluation were given a short 10 question test of requisite mathematical skills. The ten question pre-test examined high school and middle school level of mathematical proficiency. This type of test is crucial to see if students have a command of the fundamental mathematics that was assumed during the development of the software. The two questions shown below proved to be the most difficult. Nineteen percent of the students missed the first question and 34% missed the second.

1) Specify a fraction equal to .375

2) Specify a ratio larger than 5 : 2 and smaller than 20 : 6

The pretest was remarkably useful for predicting the students who would gain the most from the software and demonstrate a conceptual understanding of the material. The graph below shows student scores on the test of conceptual understanding based on their scores on the requisite or minimal skills test. While there were more remedial problems found outside of Tufts and the overall improvement slightly lower, the general pattern of improvements was the same regardless of site. Students who scored only an 8 or less (out of 10) on the minimal skills test showed only a modest gain from the software. Those students scoring a perfect 10 out of 10 showed the largest gain. It is not clear how to interpret the impact of the basic skills test. It may be that if students were effectively tutored in the areas of fractions and ratios (i.e., could answer the two most commonly missed questions), they would show much stronger gains from the software. It may also be that low scores on the minimal skills test are symptomatic of more comprehensive educational deficits.




The trace data provided two very useful results. One, it helped explain differences in class scores based on use. For example, some classes did not emphasize probability distributions and did not make much use of the probability distributions program. Large differences between two

Tufts experimental classes on probability questions (61% against 45%) became understandable when the traces revealed that students in one class used the program fairly actively and extensively, while the other class barely used the program at all.

In addition, the trace data permitted us to link student test performance to particular ways that students used the program. For instance, we found that students who investigated a probability distribution using one set of experiments did very poorly on a related question, while another group of students used a different approach and did substantially better. The students who used the better approach typically initiated it very soon after they began experimenting (median time under 2 minutes). There were other places in the program where we expected to find similar results and in deed that outcome occurred. In general, the students that stay in one spot or move very slowly are less likely to be engaged by the software and learn less from it as well.

Finally, the evaluation confimed the claim that most students leave introductory statistics courses with little concetual understanding of the material. Students in classes that did not use ConStatS averaged about 39% on the test of conceptual understanding. Even though conceptual understanding improved substantially in the classes that used ConStatS, it is clear that a major gap still exists.

As for disseminating and evaluating the project itself, we sought feedback about our own work through formal conference presentations and publications. Presenting work to a critical audience is often the best way to assess your own work. Conference presentations were very well received, often ending with listeners wanting to use some or all of the methods presented. Three of the papers on the evaluation have been published in refereed journals, while another has been included in a book on "Assessment Strategies That Work". Other papers are currently under review. We look forward to critical comments on our works in progress.

In addition to disseminating work about the evaluation, we have just signed a contract with Prentice Hall to publish ConStatS. The evaluation project offered an opportunity to showcase ConStatS, improve it, present impressive evaluation results and create new interest in it as a powerful curricular reform. Publication of the software itself should generate interest in the

evaluation as it stands today and create an opportunity to learn how ConStatS fairs at new sites.

F. Summary and Conclusions

In the new field instructional technology, it is too easy for a developer to look at their own work uncritically. The excitement and magic of seeing your ideas on a computer screen cause you to believe in their value far beyond their actual worth. The extensive, detailed nature of the evaluation forced us, as developers of the software, to look at our work with a remarkable critical eye. When we first applied to FIPSE to evaluate ConStatS, we had the a number of good indications the program would be successful. Formative evaluations had shown that students were comfortable using the software. Their attitudes were very positive -- they seemed to enjoy exchanging a class to work in computer labs and try something new. Since we had already been through one major redesign effort, and had received a fair amount of positive reaction for our design, the feeling was that we had achieved a stunning success and the evaluation would confirm it.

While the software was remarkably useful, there were several concepts students did not learn that we expected they would. These results were some of the most significant we found. For instance, on one screen, students simply see a plot of a probability distribution constructed and then select an experiment. To test this simple screen, we constructed a simple-minded question: for the normal distribution below, indicate the highest and lowest values the variable can assume. Most students get the question wrong, indicating they either did not know some very basic concepts about normal probability distributions, or they were confusing probability distributions with distributions of sample data. Results on other questions point to the latter result, and implicate misleading properties of displays as part of the problem. Had we not been dedicated to a detailed and complete evaluation of the software we may never have even asked this kind of question. Many of the questions that appeared to be the simplest, like the questions on fractions and rations, turned out to be the most informative. Likewise, some of the most handsome illustrations contain in the program, those that received the most acclaim at conference

presentations, did not successfully promote learning. It is extraordinarily difficult to anticipate what stundents think and learn when they see a graph or an illustration of a concept. They do not neccessarily see what the instructor sees, and all indications are that many instructors do not realize this. Too many student misconceptions came on questions we initially felt were too easy.

Finally, the initial hope that all students will engage in an active experimental style of learning once they experience a well-designed software package appears now to be overly optimistic. Students with remedial problems show only a marginal gain from using the software, and these students interact with the software in a slow and less focused fashion. The nature of the remedial problems are sobering since all the questions on the remedial skills examination were really testing pre-high-school levels of mathematical achievement. Surely these fundamental problems (such as not being able to do ratio/decimal transformations) will affect negatively on their performance in a wide range of university courses which require even modest levels of mathematics proficiency. However, when a student did possess the levels of mathematics proficiency expected in order to obtain admission to a university, then ConStatS usage did increase their understanding of statistics.

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement (OERI)
Educational Resources Information Center (ERIC)

# NOTICE

## REPRODUCTION BASIS

☐ This document is covered by a signed "Reproduction Release (Blanket)" form (on file within the ERIC system), encompassing all or classes of documents from its source organization and, therefore, does not require a "Specific Document" Release form.

☒ This document is Federally-funded, or carries its own permission to reproduce, or is otherwise in the public domain and, therefore, may be reproduced by ERIC without a signed Reproduction Release form (either "Specific Document" or "Blanket").